ASYMSIM: META PATH-BASED SIMILARITY WITH ASYMMETRIC RELATIONS

BY

JONATHAN CHRISTOPHER TEDESCO

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2013

Urbana, Illinois

Advisers:

Professor Jiawei Han
Lecturer Cinda Heeren

Abstract

Peer similarity search is a deceptively complex problem in information network analysis. Past research has primarily focused on similarity search in homogeneous networks, but real world data is often best represented using heterogeneous information networks, with multiple node and relation types carrying real-world semantics. Recent work addresses similarity search in heterogeneous networks by introducing the concept of *meta paths*, or paths that connect object types via a sequence of relations. These meta path-based similarity measures can capture the subtlety of peer similarity for paths containing symmetric edges, but real data contains asymmetric relations that play a significant role in peer similarity semantics, for instance citations in bibliographic networks. In this paper, we revisit the problem of peer similarity search among objects of the same type in heterogeneous information networks. We present an efficient meta path-based peer similarity measure, *AsymSim*, which both captures the semantics of peer similarity and remains sensitive to asymmetric relations in the network, allowing us to extract deeper peer semantics. We discuss how to efficiently handle *AsymSim* queries online and perform experiments on real DBLP data to verify the effectiveness of our proposed measure.

Acknowledgments

There are numerous colleagues and friends that have contributed to this work; without them, this research would not have been possible. I would like to thank my adviser Jiawei Han for his tireless work and encouragement, and my co-adviser Cinda Heeren for her discussion and support. I would like to thank Tim Weninger for his assistance obtaining the resources for running my experiments, and the research group for their co-operation while I ran many of these experiments. Finally, I would like to thank my parents and Courtney for their continuous love and support.

Table of Contents

Chapter 1

Introduction

Heterogeneous information networks consist of multi-typed objects and relations connecting them, and their intuitive, rich representation for real-world data has caused them to grow in popularity recently. However, the complexity of multiple types makes extracting data from these networks more challenging than from homogeneous networks containing only single typed objects and relations. These networks intuitively represent datasets such as bibliographic and social media networks, and it is critical to understand how to search such networks in a domain-independent way, without necessarily using the attribute data for particular objects in the network. For example, in a bibliographic network, we may be interested in finding similar authors, papers, or venues in the network based on their connectivity in the network without using domain-dependent information, such as the text similarity of their papers.

Similarity search has long been a fundamental focus in data mining, and was first introduced in the context of numerical and categorical data. Recent work such as personalized PageRank [3] and SimRank [2] proposed novel approaches to similarity search in information networks by leveraging structural links, but only in the context of single-typed, or homogeneous, information networks. Although these algorithms extend to heterogeneous networks, they do not capture the types associated with objects and relations, which carry important semantic meaning. In [9], Sun et al. propose a novel meta path-based framework for extracting data based on the multi-typed data of networks, and introduce a peer similarity search algorithm called *PathSim* that efficiently computes top-k peer similarity based on the network structure surrounding the objects.

PathSim applies to meta paths containing only symmetric relations (edges), but asymmetric relations are often valuable, such as citations in bibliographic networks. We propose *AsymSim*, which uses the notion of meta paths combined with *meta neighbors*, or neighbors along meta path instances in the network, to capture peer similarity with asymmetric relations in heterogeneous networks. Our approach thus uses a meta path-based approach to capture semantics in the network in a flexible way, allows efficient top-k search, and captures both symmetric and asymmetric relations.

The contributions of the paper are summarized as follows:

1. We propose *AsymSim* for similarity search in heterogeneous networks, capturing both asymmetric and symmetric relations.

2. We introduce an extension to the *meta path-based framework* called *meta neighbors*, that increase the descriptive power of the framework.

3. We construct *AsymSim* so that it reduces to PathSim for symmetric paths, allowing the same performance benefits and balance of visibility as PathSim.

4. We demonstrate the effectiveness of our results using demonstrative examples and case studies on the full DBLP dataset, showing AsymSim to be the first measure to capture peer similarity semantics while leveraging citation data.

In Chapter 2, we introduce previous similar work. In Chapter 3 we formalize the problem, and raise several areas for improvement over existing work. In Chapters 4 and 5, we introduce our approach and demonstrate its effectiveness on the full DBLP network. Finally, we discuss open questions and future directions for our work in Chapter 6, and conclude in Chapter 7.

Chapter 2

Related Work

Past work in data mining has addressed similarity search in the context of categorical and numerical data, but not until recently has the similarity search problem been proposed in the context of information networks.

2.1 Previous Similarity Measures

Similarity measures such as SimRank [2] and Personalized PageRank [3] represent state of the art similarity measures on homogeneous information networks. Personalized PageRank adapts the PageRank [5] algorithm using personalization vector indicating the query nodes. SimRank [2] is based on intuition that similar objects are related to similar objects, and defines a recursive iterative measure over objects in the graph. Although these measures can be extended to heterogeneous networks by ignoring the types associated with objects and relations in the network, this approach ignores the semantic meaning of different relations and paths in the graph. Further, the random walk-based nature of these measures make them biased towards highly visible objects in the network.

Measures such as PopRank [4] and ObjectRank [1] first experimented with assigning weights for measures based on types of edges in heterogeneous networks, and showed promising results. A recent measure, PathSim [9] develops this principle more fully, introducing the notion of meta paths in a heterogeneous network, or paths based on the general structure of a network that hold semantic meaning in the data. PathSim takes an alternate approach from its predecessors, considering only counts of instances of these meta paths in heterogeneous networks, rather propagating a random walk score.

2.2 Insufficiency for Asymmetric Relations

We can classify each of these measures into one of three categories: path count-based, random walk-based, or pairwise random walk-based measures.

P-PageRank, or personalized PageRank, falls into the category of a *random walk*-based measure. This approach is intuitive and clearly defined on homogeneous networks, but is biased towards highly visible objects in the network.

SimRank, a *pairwise random walk* measure, is based on the intuition that two objects are similar if they are related to similar objects. This measure is again intuitive and effective in homogeneous networks, but favors objects with skewed distributions of in versus out edges, and objects where adjacent paths meet at a small number of nodes.

PathSim addresses the problems with SimRank and P-PageRank by introducing a balance of visibility factor for each of the two nodes being compared, thus capturing the subtlety of peer similarity. However, while P-PageRank and SimRank are defined over directed networks, PathSim is only applicable to paths containing symmetric relations in heterogeneous networks. Trying to apply PathSim to asymmetric relations results in invalid similarity scores, since the normalization for the measure relies entirely on this assumption of symmetric edges.

Based on the problems with these measures, we know that an ideal heterogeneous network similarity measure on objects x and y should:

1. Be defined on heterogeneous networks to capture the semantics of multiple object and relation types connecting x and y
2. Balance scores based on the visibility of x and y, so that measures are not biased towards highly visible objects or particular structures in the graph
3. Be defined for paths containing both symmetric and asymmetric relations

To this end, we introduce *AsymSim*, a meta path-based similarity measure that captures peer relationships similar to PathSim, while preserving asymmetry in the graph.

Chapter 3

Problem Formalization

3.1 Heterogeneous Information Network

A heterogeneous information network is an information network that contains multiple types of vertices.

DEFINITION 1 Information Network. *An* information network *is defined as a directed graph* $G = (V, E)$ *with an object type mapping function* $\phi : V \rightarrow A$ *and a link type mapping function* $\psi : E \rightarrow R$*, where each object* $v \in V$ *belongs to one particular object type* $\phi(v) \in A$*, and each link* $e \in E$ *belongs to one particular relation type* $\psi(e) \in R$*.*

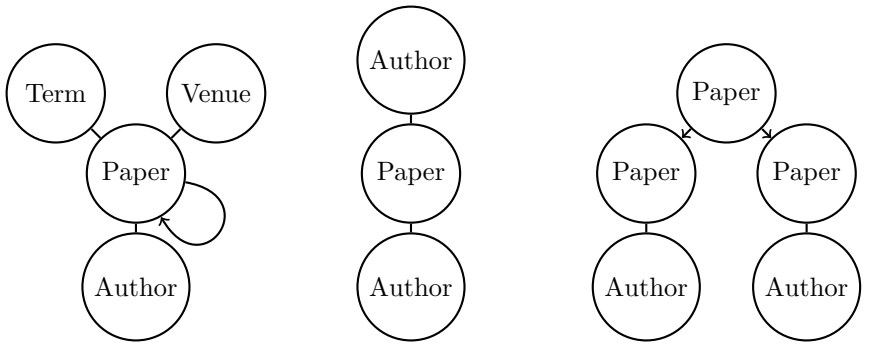
Notice that we explicitly assign types to objects and relationships individually, and that each node or edge maps to exactly one type. We notate relations between nodes based on their connecting edges, so that if nodes A and B are connected with an edge e of type R, we say ARB holds; likewise, we define the inverse of a relation, so that for ARB, $BR^{-1}A$ naturally holds. When there are multiple types of objects or relations, $|A| > 1$ or $|R| > 1$, the information network is called a **heterogeneous information network**; otherwise, we refer to it as a **homogeneous information network**.

To simplify the description of new heterogeneous information networks, we re-introduce the concept of a network schema from [9], a meta description of the network, which describes the general structure of the network using object and relation types. The network schema for an information network captures the types of objects and links can exist, defining a class of concrete information networks.

DEFINITION 2 Network Schema. *A* network schema *is schema description defining a template for an information network* $G = (V, E)$ *with object type mapping* $\phi : V \rightarrow A$ *and link mapping* $\psi : E \rightarrow R$*, and is a directed graph defined over object types* A *and relations* R*, denoted as* $T_G = (A, R)$*.*

Using this notion of network schema, we can describe a class of information network. Consider a **bibliographic information network**, as introduced in [9] and described in the following example.

EXAMPLE 1 *A* **bibliographic information network** *is a heterogeneous network containing four types of objects: papers (P), venues (conferences or journals) (C), authors (A), and terms (T). These object types and their potential relations are shown in the network schema in Figure 3.1a. Links exist between authors and papers representing* writing *or* written-by *relations, between venues and papers representing* publishing *or* published-in *relations, between papers and terms representing* using *or* used-by *relations, and between papers representing* citing *or* cited-by *relations.*



(a) Network Schema (b) APA Meta Path (c) $AP \leftarrow P \rightarrow PA$ Meta Path

Figure 3.1: Bibliographic Network Schema and Meta Paths

When discussing the semantics of these heterogeneous networks, **objects** in the connected data correspond to **vertices** in the network, and **relations** between objects in the data correspond to **edges** in the network. For example, in DBLP, authorship connects author and paper objects together, and we say it is *symmetric*, since it does not make sense to discuss authorship in a particular direction. Conversely, the citation relation is an *asymmetric* relation connecting two paper objects, since citations between papers appear in only one direction.

3.2 Meta Path-Based Similarity Framework

In information networks, two objects are often connected by multiple paths in the network. In heterogeneous information networks, these paths may contain various semantic meanings, based on the types of nodes along the path. For example, in a bibliographic information network, two authors may be connected via an *author-paper-author* path, representing coauthorship, *author-paper-venue-paper-author* path, representing publishing in the same conference, and so on. We formally define the semantic meaning of these paths by considering paths in the network as instances of particular *meta paths*, defined below.

DEFINITION 3 Meta path. *A meta path P is a path defined on the graph of the network schema $T_G = (A, R)$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$. This defines a composite relation $R = R_1 \circ R_2 \circ \cdots \circ R_l$ between type A_1 and A_{l+1}, where $\circ$ denotes the composition operator on relations.*

We say that a path p in a particular network follows some meta path P if the ordered nodes and edges of p match the object types and relations of P. Formally, a path $p = (v_1 v_2 v_3 \cdots v_k)$ in a particular network follows some meta path P if $\forall i, \phi(v_i) = A_i$, and $\forall i, e_i = \langle v_i v_{i+1} \rangle, \psi(e_i) = R_i$.

The **length** of a meta path P is the number of relations in P, and is **symmetric** if the relation defined by P is symmetric, that is if R is equivalent to R^{-1}. Note that for meta path relation $R = R_1 \circ R_2 \circ \cdots \circ R_l$, the symmetry of R is not necessarily related to the symmetry of any individual relation R_i.

R may be symmetric even if some relation R_i is not. Consider bibliographic coupling for two authors as an example, represented in the DBLP network by meta path $AP \rightarrow P \leftarrow PA$, or co-

citation, represented by $AP \leftarrow P \rightarrow PA$. Although both of these meta paths contain asymmetric relations, the composite relation between authors defined by the meta path is symmetric. Figure 3.1b shows the co-authorship meta path, APA, a symmetric meta path containing only symmetric relations, while Figure 3.1c illustrates the co-citation meta path, which is symmetric but contains asymmetric relations.

We refer to paths following P in an information network G as **path instances** of P in G, and define the **reverse meta path** P^{-1} of P to be the meta path defined by R^{-1}.

DEFINITION 4 Meta neighbor. *A set of meta neighbors $N_{x,d,P}$ represents the set of all objects in the network that connect to object x along instance of meta path P in direction d. For example, $N_{x,in,P} = \{y|p = yRx \in P\}$ represents the set of objects y where at least one meta path instance exists in the network that starts at y and ends at x, $y \in A_1$ and $x \in A_{l+1}$.*

To simplify the notation of meta paths, we use type names denoting the meta path if the resulting path is unambiguous. For example, in the bibliographic network, the co-author relation can be described using the meta path A–P–A, or APA for short if there is no ambiguity. We likewise shorten paths containing asymmetric relations, as long as the shortened version is also unambiguous. For example, we may shorten A–$P \rightarrow P$–A to $AP \rightarrow PA$, since the *authorship* relation (A–P) is always symmetric.

Given a user-specified meta path P, a similarity measure can be defined for a pair of objects $x \in A_1$, $y \in A_l$ given P. In general, we define a meta path-based similarity framework for objects x and y on a meta path as: $s(x, y, P) = \Sigma_{p \in P} f(p)$, where P is a particular meta path defined on network schema T_G, p is a particular path instance of P defined on network G, and $f(p)$ is some function on a single path instance p. Since we only focus on peer similarity, we only consider measures for same-typed nodes in the network, making the assumption that $A_1 = A_l$ for any meta path $P_i = (A_1 R_1 A_2 R_2 \cdots R_{l-1} A_l)$. Although an individual relation R_i of some path may be asymmetric, we only address symmetric relations $R = R_1 \circ R_2 \circ \cdots \circ R_l$.

Chapter 4

AsymSim

In this section, we introduce *AsymSim*, including its intuition, examples, and relationship to existing similarity measures.

4.1 AsymSim: Asymmetric Similarity Measure

Motivated by problems with existing peer similarity search measures, we propose *AsymSim*, a meta path-based peer similarity measure on heterogeneous networks. Our intuition behind this measure is that two objects should be connected and have similar visibility in the network, and that we should be able to define this measure for meta paths containing both symmetric and asymmetric relations. However, since peer similarity should be symmetric between two nodes x and y, we allow asymmetric relations within a path, but still only allow symmetric paths, thus defining a symmetric measure between the two objects.

DEFINITION 5 ***AsymSim****. Given symmetric meta path $P = Q^{-1}Q$, AsymSim between x and y along P is defined as:*

$$s(x, y, P) = \frac{2 \sum_{z \in N_{in,Q}} |p_{z \rightsquigarrow x}||p_{z \rightsquigarrow y}|}{\sum_{z \in N_{x,in,Q}} |p_{z \rightsquigarrow x}|^2 + \sum_{z \in N_{y,in,Q}} |p_{z \rightsquigarrow y}|^2} \tag{4.1}$$

where $p_{z \rightsquigarrow x}$ represents the set of path instances of Q from z to x in the network, $N_{x,in,Q}$ represents the in-neighbors to x in the network along instances of Q, and $N_{in,Q} = \{N_{x,in,Q} \cap N_{y,in,Q}\}$.

The numerator captures the contextual similarity of the two nodes in the graph. Specifically, the numerator of this measure is based on the number of path instances from shared in neighbors of x and y. This value is the product of the number paths to x and to y, for each shared in-neighbor z of x and y (twice the sum of this value for each z). The numerator thus captures the path instances of P connecting x and y. The denominator normalizes by the visibility of each node x and y, by looking at the number of paths from each in-neighbor of x to x, and similarly for y (the sum of the counts squared, for each in-neighbor z).

Consider a toy example with eight authors and two research areas, say *data mining* and *databases*, with once conference each, *KDD* and *VLDB* respectively. In our example, three of these eight authors are *multi-disciplinary* authors, having published papers in both research areas, while the rest have published papers in only one or the other. In Table 4.1a, we show the number of papers published by each author in each conference, represented by the number of APC meta path instances connecting the author and conference nodes in the network. In Table 4.1b, we show the total number

Author	A	B	C	**D**	**E**	F	G	H	I
VLDB	67	71	82	**46**	**49**	0	0	0	**48**
KDD	0	0	0	**47**	**45**	66	69	86	**49**

(a) Publication Count for Each Author by Conference

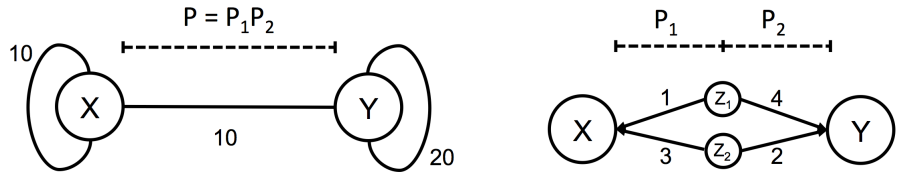
Author	**A**	B	C	**D**	**E**	**F**	G	H	I
Citations	**104**	89	15	**138**	**111**	**107**	54	15	38

(b) Total Citations for Each Author

Author	A	B	C	**D**	**E**	F	G	H	**I**
PathSim Score ($APCPA$)	0.70	0.70	0.68	**1**	**1**	0.71	0.71	0.69	**1**
AsymSim ($AP \leftarrow PCP \rightarrow PA$)	0.72	0.72	0.22	**1**	**0.98**	0.69	0.69	0.21	**0.52**

(c) PathSim And AsymSim Similarity Scores

Table 4.1: Multi-Disciplinary Authors Example



(a) *PathSim* Perspective (b) *AsymSim* Perspective

Figure 4.1: Illustration of *PathSim* and *AsymSim* for Symmetric Paths

of citations for each author, or equivalently, the number of incoming $P \rightarrow PA$ meta path instances to each author.

In our example, we consider the problem of finding the most similar author to D. Note that we have three multi-disciplinary authors in this example: D, E, and I. A, E, and F are the authors with the most similar citation counts in the network. Intuitively, if we consider both the publication record and citation record of the authors, we would say that E is the most similar author to D, since E is the only author similar in both ways.

In Table 4.1c, we see the similarity scores according to *PathSim* with meta path $APCPA$. We can see that it clearly distinguishes multi-disciplinary authors, but does not distinguish based on the reputation (citation count) of authors, since it evaluates E and I to be equally similar to D. On the other hand, we show the results using *AsymSim* with meta path $AP \leftarrow PCP \rightarrow PA$, which captures conferences with papers that cite both authors. This path is symmetric, but contains asymmetric relations, and captures both the publication and citation records of each author. In Table 4.1c, we see that E is found to be the most similar author to D using this path with *AsymSim*, which matches our intuition.

In the following sections, we further explain the intuition behind this measure and its relations to PathSim, including how to efficiently compute this measure.

	SIGMOD	VLDB	ICDE	KDD
Mike	2	1	0	0
Jim	50	20	0	0
Mary	2	0	1	0
Bob	2	1	0	0
Ann	0	0	1	1

(a) Path Instances for APC Meta Path

	Jim	Mary	Bob	Ann
PathSim	0.0826	**0.8**	**1**	0
AsymSim	0.0826	**0.8**	**1**	0

(b) Similarity To 'Mike'

Table 4.2: DBLP Authors Example [9]

4.2 PathSim Derivation

For the case of paths containing only symmetric relations, AsymSim is equivalent to the PathSim, since *AsymSim* simply computes *PathSim* by between x and y by calculating the count of path instances through each midpoint object along $P = P_1P_2$. In Figure 4.1, we see an illustration of the shift in perspective from PathSim to AsymSim. Consider the PathSim perspective for path $P = P_1P_2$ in Figure 4.1 with nodes X and Y. In this example, we have 10 path instances between X and Y, 10 path instances from X back to X, and 20 instances from Y back to Y. So, we would compute the PathSim similarity score $s(x, y) = \frac{2(10)}{10+20} = \frac{2}{3}$.

Alternatively, we could consider the midpoint objects along each path instance of P, looking at path instances of P_1 and P_2 to neighbors of X and Y. Without loss of generality, suppose that we have two nodes Z_1 and Z_2 along P, connected to X and Y as shown in Figure 4.1. Then, we can calculate the total number of instances of P by counting the total number of possible paths from X to Y along P_1P_2, i.e. through Z_1 or Z_2. In our example, this yields $1 \cdot 4$ instances of P through Z_1 and $3 \cdot 2$ instances of P through Z_2, for a total $(1 \cdot 4) + (3 \cdot 2) = 10$ path instances of P between X and Y.

If we assume the network is setup as shown in Figure 4.1, all cycles including X or Y must pass through Z_1 or Z_2, and so we can directly count the number of possible cycles from X and Y by counting all possible combinations of paths from X or Y to Z_1 or Z_2 and back. Using this strategy, we get $4 \cdot 4 = 16$ cycles for Y through Z_1 and $2 \cdot 2 = 4$ cycles for Y through Z_2, for $4^2 + 2^2 = 20$ self loops for Y. Similarly, we count $1 + 3 \cdot 3 = 1^2 + 3^2 = 10$ self loops for X. Thus, we get a *AsymSim* score of $\frac{2(1 \cdot 4 + 3 \cdot 2)}{(4^2 + 2^2) + (1^2 + 3^2)} = \frac{20}{30} = \frac{2}{3}$.

Likewise, consider the DBLP graph example in Table 4.2a, as shown in [9]. In Table 4.2b, where we compute PathSim on the meta path $APCPA$, and AsymSim using the path $APCPA$ by counting CPA path instances from shared conference meta neighbors to the two author objects in the network. Since each relation in the meta path is symmetric, we compute identical similarity measures to Mike as shown in [9].

4.3 Commuting Matrix Computation

Let us discuss how we can compute *AsymSim* on a real data set. First, we introduce the concept of a *commuting matrix* [9]:

DEFINITION 6 ***Commuting matrix.*** *Given a network $G = (V, E)$ and meta path P, a*

commuting matrix M for meta path $P = (A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1})$ is defined as $M = W_{R_1} W_{R_2} \cdots W_{R_l}$, where W_{R_i} is the adjacency matrix for relation R_i, between types A_i and A_{i+1}.

Thus, for two objects x and y along meta path P, the count 'in-neighbors' of x and y along instance of P correspond to the counts in columns i and j in M. We refer to these columns as v_i and v_j, for x and y respectively, and use dot products involving this vectors to calculate the *AsymSim* measure between x and y. So, we compute *AsymSim* using M_P, with meta path $Q = P^{-1}P$, as $s(x, y, Q) = \dfrac{2v_i v_j}{v_i v_i + v_j v_j}$.

To compute the most similar objects to a given object using the *AsymSim* measure, we must be able to executed top-k queries efficiently at query time. For longer meta paths, computing meta path instances between objects at run-time becomes impractical to store for large networks [9].

To address this problem, for a longer meta path P, we can decompose P into shorter meta paths $P = P_1 P_2$, and store commuting matrices for P_1 and P_2 (matrices M_1 and M_2), rather than for P directly. Then, we can compute M for P at query time, by taking the product of matrices M_1 and M_2, so $M = M_1 M_2$.

Using this practice, we can decompose long meta paths into any number of commuting matrices that store shorter meta paths. To reduce the space complexity for storing each of these matrices, we can use sparse matrices, only materializing the sparse matrix M at query time, which will be dense compared to the smaller individual matrices corresponding to shorter meta paths within P.

4.4 Path Length Generalization

Likewise, these meta path-based approaches are rigid in terms of the path length. Although we can control the particular path to use, and weight together multiple measures using various paths, these meta path-based metrics are not well-suited to capturing various length paths at their core. For example, it is difficult to quantify the relative importance of the meta path APA versus $(APA)^2 = APAPA$ versus $(APA)^3 = APAPAPA$.

We make two observations about our intuition for longer paths. First, a fixed, short length path will certainly return the most similar nodes in a network, but may miss objects in the network that are slightly beyond the chosen path. On the other hand, the longer a meta path becomes, the less meaningful its similarity results may become. Intuitively, in the DBLP network, two authors may still be similar if they are connected by the $(APA)^2 = APAPA$ path but not by the APA path. However, for longer paths, such as $(APA)^3$ or $(APA)^{10}$, similarity is propagated to remote neighborhoods in the network, and we introduce many 'similar' objects that are not semantically meaningful.

We propose balancing these opposing forces by defining similarity measures along arbitrarily long meta paths, discounting the contribution of longer meta paths to the total score based on their length with respect to the original meta path.

Specifically, for meta path P, we propose two iterative measures, where k is the maximum number of iterations, or maximum k for P^k:

1. **Constant**, or $AsymSim_C$. Discount the value of the meta path P^i by C^i, for constant $C \in [0, 1]$:

Author	D	E	I	J
$AsymSim_C$, $k = 1$	1	0.960	0.380	0
$AsymSim_C$, $k = 2$	1	0.960	0.380	0.120
$AsymSim_C$, $k = 3$	1	0.960	0.380	0.220
$AsymSim_C$, $k = 4$	1	0.960	0.380	0.270
$AsymSim_C$, $k = 50$	1	0.960	0.380	0.310

(a) $AsymSim_C$ with $C = 0.5$, $P = (AP \leftarrow PAP \rightarrow PA)$

Author	D	E	I	J
$AsymSim_{PA}$, $k = 1$	1	0.960	0.380	0
$AsymSim_{PA}$, $k = 2$	1	0.950	0.310	0.120
$AsymSim_{PA}$, $k = 3$	1	0.940	0.280	0.150
$AsymSim_{PA}$, $k = 4$	1	0.940	0.270	0.190
$AsymSim_{PA}$, $k = 50$	1	0.940	0.270	0.240

(b) $AsymSim_{PF}$ with $C = 0.5$, $P = (AP \leftarrow PAP \rightarrow PA)$

Figure 4.2: *AsymSim* Similarity for Generalized Paths

$$s(x, y, P) = AsymSim_C(x, y, P) = \sum_{i=1}^{k} C^{i-1} AsymSim(x, y, P^i) \tag{4.2}$$

2. **Preferential Attachment**, or $AsymSim_{PA}$. Discount the value of the meta path P^i by both a constant C^i, $C \in [0, 1]$ and the similarity score of the previous iteration:

$$s(x, y, P) = AsymSim_{PA}(x, y, P) = \sum_{i=1}^{k} C^{i-1} AsymSim(x, y, P^{i-1}) AsymSim(x, y, P^i) \tag{4.3}$$

In the numerator for each measure, we weight the similarity along various length repetitions of some meta path, and the denominator for each normalizes the final measure such that $s(x, y, P) \in [0, 1]$.

Consider the network described in Figure 4.1, but let us introduce another multi-disciplinary author J identical publications to D, E, and I, and the same total number of citations as D. Suppose J is cited only by authors E and I, but it not cited by D directly.

If we consider the citation meta path, $AP \rightarrow PA$, since there are no such paths connected D to J, our measure would say that D and J have a similarity of 0. However, D and J are highly connected along the $(AP \rightarrow PA)^2$ meta path, although they have no connections along $AP \rightarrow PA$, and although this is not as significant as being directly connected via the $AP \rightarrow PA$ path, we would intuitively say that there is some nonzero similarity.

In Figure 4.2, we show the similarity according to these two measures for the subset of multi-disciplinary authors for this network. For each example, we look at several iterations of the measures ($k = 1, 2, 3, 4, 50$), for a meta path $P = AP \rightarrow PA$. Using $AsymSim_C$, we see that the similarity of J to D increases through the first several iterations and levels off, without significantly affecting the scores of other objects in the network. On the other hand, $AsymSim_{PA}$ also increases the score of J during the first several iterations, but is slower to level off. Since this measure is based on the measure of the previous iteration, the scores of less similar nodes change more than the scores of highly similar nodes, effectively smoothing the scores of lower scoring nodes.

Although further study is necessary to choose a suitable C value, this approach shows promise for lessening the rigidity of fixed-length meta path similarity measures, and only requires only a fixed number k iterations to calculate.

Chapter 5

Experiments

5.1 Experimental Setup

We use the "four area dataset" introduced in [9], based on a DBLP dataset downloaded in 2009 and clustered into four research areas: *database*, *data mining*, *machine learning*, and *information retrieval* [8]. This dataset contains 20 conferences, $28K$ authors, $28K$ papers, and $13K$ terms extracted from the DBLP dataset, representing top conferences, authors, and papers from each area. We remove stop words from paper titles and terms, and perform stemming to further reduce the number of nodes in the network. Further, we add citations for this subset from a newer Arnetminer 2011 DBLP dataset [13] [10] [12] [11], which adds $46K$ citations between the papers in our network.

5.2 Similar Authors in DBLP

Let us revisit the problem of finding similar peer authors in the DBLP information network. Intuitively, we consider two authors to be similar if they publish similar work and have similar 'visibility' in the network. Further, we intuitively attach significance to the number of times an author has been cited, since this generally correlates well to their 'reputation' in their research area. Thus, similar authors should also have similar numbers of citations.

In the DBLP network, these characteristics manifest themselves through meta path instances in the network. In previous work, we consider meta paths with only symmetric relations in the network, which captures peer similarity based on research area and number of publications, but overlooks citation information. Using *AsymSim*, we capture more information by using both symmetric and asymmetric meta paths in the network.

Consider the question 'Who is most similar to Christos Faloutsos?'. Intuitively, we would say authors in the same research area, with similar publication record, and who have been cited similarly in the network. In Table 5.1, we show the top 10 most similar authors according to the $APCPA$ meta path using *PathSim*, showing the total citation count of each author in our network. This meta path captures the number of times authors are published in the same conferences. Although we see that the results are intuitive based on the research areas and number of publications of each author, the number of citations, or 'reputation' of the authors in the research area are not necessarily very similar. For example, the citation counts of *Rakesh Agrawal* and *Christos Faloutsos* are very dissimilar, but they are considered very similar by PathSim, since their publication records are similar.

On the other hand, in Table 5.2, we show the similarity results for Christos Faloutsos using the

Rank	Author	Citations	Publications	PathSim Score
1	**Christos Faloutsos**	**634**	**127**	**1**
2	Jiawei Han	665	168	0.906
3	Rakesh Agrawal	1511	105	0.901
4	Hans-Peter Kriegel	617	102	0.839
5	Jian Pei	270	70	0.831
6	Raghu Ramakrishnan	652	95	0.809
7	H. V. Jagadish	546	106	0.804
8	Nick Koudas	351	79	0.788
9	Hector Garcia-Molina	613	98	0.779
10	Divesh Srivastava	464	103	0.775

Table 5.1: Similarity to *Christos Faloutsos* Using *PathSim* with $APCPA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Christos Faloutsos**	**634**	**127**	**1**
2	Hans-Peter Kriegel	617	102	0.982
3	Rajeev Motwani	542	36	0.963
4	Raghu Ramakrishnan	652	95	0.957
5	H. V. Jagadish	546	106	0.944
6	Divesh Srivastava	464	103	0.924
7	Jennifer Widom	726	64	0.919
8	Yannis E. Ioannidis	439	50	0.913
9	Surajit Chaudhuri	581	96	0.912
10	Hamid Pirahesh	439	47	0.910

Table 5.2: Similarity to *Christos Faloutsos* Using *AsymSim* with $AP \leftarrow PCP \rightarrow PA$ meta path

$AP \leftarrow PCP \rightarrow PA$ meta path with AsymSim, which captures the number of times authors are cited by papers in the same conferences. Although the list is similar to the results of *PathSim* using $APCPA$, we see that authors with dissimilar citation counts are discounted, and so the overall results are more intuitive.

Note that this similarity measure does not directly measure the similarity of the citation counts of authors, but rather calculates paths throuigh conference papers that cite both authors. While this path helps capture the citation count of authors, certain authors that have similar citation counts to Christos Faloutsos may not appear as one of the most similar authors under this experiment if they are not similarly cited from the same conferences. For example, in Table 5.1, Jiawei Han appears in the top most similar authors as Christos Faloutsos, and has a similar citation count. However, Han does not appear in the top most similar authors under the AsymSim experiment in Table 5.2, indicating that although the two authors are published in the same conferences and have a similar citation count, they do not have as high of a score as authors that were cited similarly from the same conference papers. This subtlety is captured by the meta path chosen, and so is configurable by the user.

Just as in the meta path-based framework for PathSim, adjusting the meta paths for AsymSim

Rank	Author	Citations	Publications	AsymSim Score
1	**Christos Faloutsos**	**634**	**127**	**1**
2	Hans-Peter Kriegel	617	102	0.579
3	Bernhard Seeger	389	36	0.546
4	Nick Roussopoulos	336	35	0.506
5	H. V. Jagadish	546	106	0.436
6	Ralf Schneider	242	4	0.426
7	Norbert Beckmann	210	1	0.391
8	Timos K. Sellis	230	28	0.389
9	Dimitrios Gunopulos	264	46	0.297
10	Philip S. Yu	556	216	0.291

Table 5.3: Similarity to *Christos Faloutsos* Using *AsymSim* with $AP \leftarrow PAP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Christos Faloutsos**	**634**	**127**	**1**
2	Hans-Peter Kriegel	617	102	0.914
3	Bernhard Seeger	389	36	0.831
4	H. V. Jagadish	546	106	0.831
5	Nick Roussopoulos	336	35	0.807
6	Ralf Schneider	242	4	0.719
7	Jeffrey F. Naughton	801	85	0.709
8	Divesh Srivastava	464	103	0.673
9	Kyuseok Shim	315	32	0.667
10	Jennifer Widom	726	64	0.666

Table 5.4: Similarity to *Christos Faloutsos* Using *AsymSim* with $AP \leftarrow PTP \rightarrow PA$ meta path

results in drastically different similarity semantics in the network. Consider the same problem addressed with the $AP \leftarrow PAP \rightarrow PA$ or $AP \leftarrow PTP \rightarrow PA$ paths, which are not sensitive to the conference, but rather capture only when two authors have been cited by the same authors or papers on the same subject. These examples are shown in Tables 5.3 and 5.4, respectively. Using a path such as $AP \leftarrow P \rightarrow PA$, the results for which are shown in Table 5.5, captures when two authors are cited by the same paper. This approach is more rigid, and we see the intuitive difference in the results than for other paths.

Consider another case study, with author 'Sergey Brin'. This author published relatively few papers, but was cited many more times than authors with a similar number of publications. In Table 5.6, we see the most similar authors computed using the $APCPA$ meta path with *PathSim*. Note that authors considered similar using this measure have a significantly different citation count than Brin.

In Table 5.7, we see the similarity scores for this same author using AsymSim with the $AP \leftarrow PCP \rightarrow PA$ meta path, which represents conferences with papers citing both authors. This captures authors that are both in the same research area and have a similar *reputation* in the network. In Table 5.7, we see that similar authors returned using this method have both similar citation and

Rank	Author	Citations	Publications	AsymSim Score
1	**Christos Faloutsos**	**634**	**127**	**1**
2	Nick Roussopoulos	336	35	0.452
3	Hans-Peter Kriegel	617	102	0.451
4	Timos K. Sellis	230	28	0.406
5	Bernhard Seeger	389	36	0.402
6	Ralf Schneider	242	4	0.350
7	Norbert Beckmann	210	1	0.330
8	Ibrahim Kamel	88	10	0.321
9	H. V. Jagadish	546	106	0.287
10	Stefan Berchtold	110	12	0.197

Table 5.5: Similarity to *Christos Faloutsos* Using *AsymSim* with $AP \leftarrow P \rightarrow PA$ meta path

Rank	Author	Citations	Publications	PathSim Score
1	**Sergey Brin**	**166**	**7**	**1**
2	Nicola Onose	6	7	0.944
3	Larry Kerschberg	10	7	0.933
4	Ion Stoica	10	5	0.929
5	Yannis Vassiliou	35	5	0.929
6	Boon Thau Loo	28	5	0.929
7	Fatma Özcan	24	6	0.914
8	Xin Dong	21	6	0.914
9	Abhijit Pol	7	6	0.914
10	Chun Zhang	181	6	0.914

Table 5.6: Similarity to *Sergey Brin* Using *PathSim* with $APCPA$ meta path

publication counts to Brin.

Capturing the citation counts of authors in similarity search has practical significance, but is overlooked by using only symmetric paths in the network. Taking advantage of a single asymmetric path, we are able to distinguish between ‘leaders‘ and ‘followers‘ in the network. Although this does not directly consider the number of papers published by each author, we can combine information from these new meta paths and from symmetric paths to capture richer data for authors in the DBLP network.

Again, we see that the semantics clearly vary significantly based on the path chosen for this author. In Tables 5.9 and 5.8, we see authors that are cited by the same authors and cited by the same papers, respectively. These paths, $AP \leftarrow PAP \rightarrow PA$ and $AP \leftarrow PTP \rightarrow PA$, capture research area without necessarily enforcing that the authors publish in the same particular conferences. However, since an author is likely to cite the most reputable of any sources that address a particular topic, $AP \leftarrow PAP \rightarrow PA$ in particular introduces subtle semantics.

As shown with ‘Christos Faloutsos’, the $AP \leftarrow P \rightarrow PA$, shown in Table 5.10, is more rigid and noisy, looking at papers that cite both authors. Since a particular paper often only cites a small number of papers that address a particular research area, this path may be biased towards papers

Rank	Author	Citations	Publications	AsymSim Score
1	**Sergey Brin**	**166**	**7**	**1**
2	Heikki Mannila	131	63	0.953
3	Roberto J. Bayardo Jr.	142	16	0.946
4	Hannu Toivonen	120	18	0.929
5	Prabhakar Raghavan	195	24	0.924
6	Eamonn J. Keogh	142	55	0.910
7	Mohammed Javeed Zaki	116	32	0.901
8	Wei Wang	135	83	0.898
9	Ke Wang	114	55	0.889
10	Tian Zhang	127	4	0.884

Table 5.7: Similarity to *Sergey Brin* Using *AsymSim* with $AP \leftarrow PCP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Sergey Brin**	**166**	**7**	**1**
2	Hannu Toivonen	120	18	0.884
3	Jong Soo Park	94	4	0.816
4	Heikki Mannila	131	63	0.800
5	Edward Omiecinski	93	21	0.800
6	Roberto J. Bayardo Jr.	142	16	0.782
7	Craig Silverstein	84	5	0.780
8	Shamkant B. Navathe	114	38	0.771
9	Ashok Savasere	68	3	0.769
10	Shalom Tsur	96	12	0.757

Table 5.8: Similarity to *Sergey Brin* Using *AsymSim* with $AP \leftarrow PTP \rightarrow PA$ meta path

that write high quality papers, or are considered 'leaders' in their areas.

Numerous additional case studies for author similarity are provided in Appendix A. These examples include many of the same meta paths shown in this section, for highly visible authors 'Rakesh Agrawal' and 'Philip S. Yu', as well as a less widely published author, 'AnHai Doan'.

5.3 Similar Papers in DBLP

Consider another challenging similarity search problem in the DBLP network: finding similar papers to a given paper. Intuitively, we consider a paper to be similar if it deals with the same subject, and is similarly influential in the research area.

In the DBLP network, two papers may be considered similar if they are published in the same conference (PCP), are written by the same author (PAP), or contain the same terms (PTP). In the DBLP network, each paper is associated with exactly one conference, and a small number of authors, but many terms, and so the PTP provides the most meaningful results for finding similar papers, of these three paths.

However, the number of citations a paper receives is an important indicator if its influence in the

Rank	Author	Citations	Publications	AsymSim Score
1	**Sergey Brin**	**166**	**7**	**1**
2	Craig Silverstein	84	5	0.817
3	Hannu Toivonen	120	18	0.645
4	Jong Soo Park	94	4	0.583
5	Shalom Tsur	96	12	0.568
6	Tomasz Imielinski	343	30	0.562
7	Alex Pang	57	2	0.560
8	Heikki Mannila	131	63	0.528
9	Arun N. Swami	404	15	0.516
10	Yiwen Yin	73	3	0.487

Table 5.9: Similarity to *Sergey Brin* Using *AsymSim* with $AP \leftarrow PAP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Sergey Brin**	**166**	**7**	**1**
2	Craig Silverstein	84	5	0.676
3	Shalom Tsur	96	12	0.500
4	Rajeev Motwani	542	36	0.412
5	Ashok Savasere	68	3	0.297
6	Jeffrey D. Ullman	438	43	0.289
7	Tomasz Imielinski	343	30	0.280
8	Arun N. Swami	404	15	0.267
9	Hannu Toivonen	120	18	0.265
10	Edward Omiecinski	93	21	0.261

Table 5.10: Similarity to *Sergey Brin* Using *AsymSim* with $AP \leftarrow P \rightarrow PA$ meta path

Rank	Paper	Citations	PathSim Score
1	**Mining Association Rules between Sets of Items in Large Databases**	**258**	**1.0**
2	The Rough Set Approach to Association Rule Mining	0	0.722
3	Mining Association Rules in Hypertext Databases	0	0.71
4	Sampling Large Databases for Association Rules	50	0.706
5	Ordinal Association Rules for Error Identification in Data Sets	1	0.703
6	Association Rules in Incomplete Databases	0	0.688
7	Mining Association Rules with Item Constraints	0	0.688
8	Fast Algorithms for Mining Association Rules in Large Databases	305	0.686
9	Discovering Association Rules in Large, Dense Databases	0	0.667
10	Association Rules	0	0.667

Table 5.11: Similarity to *Mining Association Rules between Sets of Items in Large Databases* Using *PathSim* with PTP meta path

Rank	Paper	Citations	AsymSim Score
1	**Mining Association Rules between Sets of Items in Large Databases**	**258**	**1.0**
2	Fast Algorithms for Mining Association Rules in Large Databases	305	0.969
3	An Effective Hash Based Algorithm for Mining Association Rules	60	0.628
4	Mining Quantitative Association Rules in Large Relational Tables	65	0.575
5	An Efficient Algorithm for Mining Association Rules in Large Databases	57	0.543
6	Discovery of Multiple-Level Association Rules from Large Databases	46	0.503
7	Dynamic Itemset Counting and Implication Rules for Market Basket Data	57	0.447
8	Finding Interesting Rules from Large Sets of Discovered Association Rules	48	0.432
9	Sampling Large Databases for Association Rules	50	0.425
10	Beyond Market Baskets: Generalizing Association Rules to Correlations	54	0.397

Table 5.12: Similarity to *Mining Association Rules between Sets of Items in Large Databases* Using *AsymSim* with $P \leftarrow PTP \rightarrow P$ meta path

Rank	Paper	Citations	AsymSim Score
1	**Mining Association Rules between Sets of Items in Large Databases**	**258**	**1.0**
2	Fast Algorithms for Mining Association Rules in Large Databases	305	0.986
3	BIRCH: An Efficient Data Clustering Method for Very Large Databases	126	0.717
4	Mining Sequential Patterns	106	0.712
5	Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications	89	0.626
6	Mining Frequent Patterns without Candidate Generation	73	0.561
7	Text Categorization with Suport Vector Machines: Learning with Many Relevant Features	161	0.553
8	The R*-Tree: An Efficient and Robust Access Method for Points and Rectangles	210	0.534
9	Mining Quantitative Association Rules in Large Relational Tables	65	0.472
10	A Comparative Study on Feature Selection in Text Categorization	112	0.47

Table 5.13: Similarity to *Mining Association Rules between Sets of Items in Large Databases* Using *AsymSim* with $P \leftarrow PCP \rightarrow P$ meta path

Rank	Paper	Citations	AsymSim Score
1	**Mining Association Rules between Sets of Items in Large Databases**	**258**	**1.0**
2	Fast Algorithms for Mining Association Rules in Large Databases	305	0.684
3	Mining Sequential Patterns	106	0.465
4	Beyond Market Baskets: Generalizing Association Rules to Correlations	54	0.457
5	An Effective Hash Based Algorithm for Mining Association Rules	60	0.438
6	BIRCH: An Efficient Data Clustering Method for Very Large Databases	126	0.427
7	Mining Frequent Patterns without Candidate Generation	73	0.417
8	Discovery of Multiple-Level Association Rules from Large Databases	46	0.394
9	Exploratory Mining and Pruning Optimizations of Constrained Association Rules	47	0.39
10	Mining Quantitative Association Rules in Large Relational Tables	65	0.384

Table 5.14: Similarity to *Mining Association Rules between Sets of Items in Large Databases* Using *AsymSim* with $P \leftarrow PAP \rightarrow P$ meta path

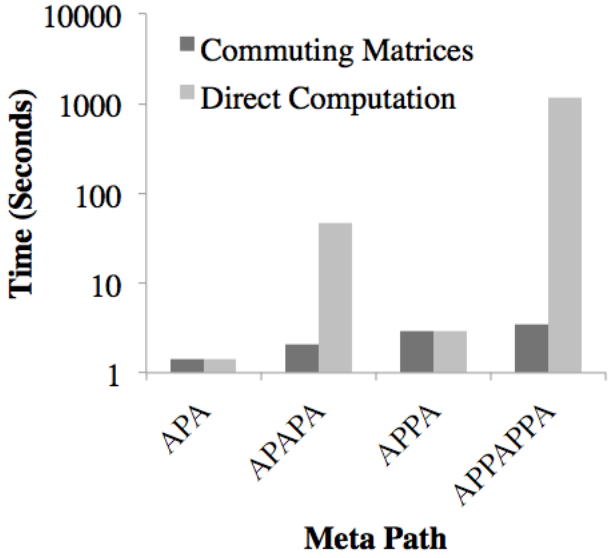
research area, and cannot be captured by these paths. Meta paths with only symmetric relations cannot capture this citation relationship, since citations are represented as asymmetric $P \rightarrow P$ meta paths in the network.

For example, consider a highly cited paper in our subset of the DBLP network, called *Mining Association Rules between Sets of Items in Large Databases*. In Table 5.11, we show the most similar papers to this paper in the network, according to the PTP meta path using *PathSim*. Although we see that the most similar papers deal with the same topics, many papers that have few or no citations are returned. On the other hand, consider the results from using *AsymSim* with the $P \leftarrow PTP \rightarrow P$ meta path, which captures papers that are cited by papers with the same terms. These results are shown in Table 5.12, and we clearly see that not only are papers covering similar topics returned, but these papers have similar citation counts to the query paper. Thus, these results match our intuition much more closely.

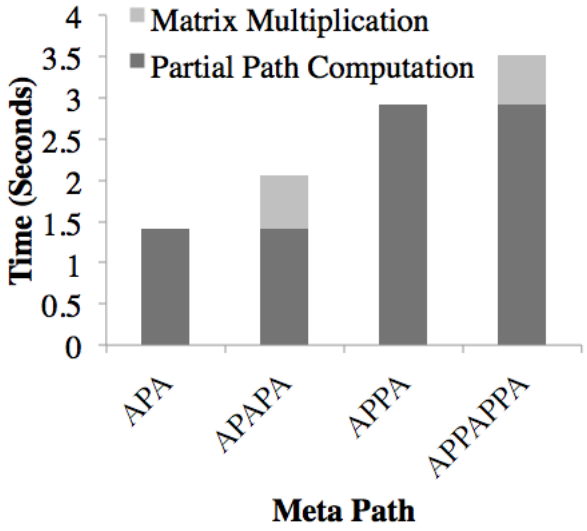
With this same query paper, we may adjust the query meta path to capture different semantics in the network. In Table 5.13, we show the most similar papers according to the $P \leftarrow PCP \rightarrow P$ meta path, which captures how often two papers are cited from the same *conference*, which is more restrictive and may indicate additional information such as the quality of the paper, based on the quality of the conference. Likewise, Table 5.14 shows the most similar papers according to authors that cite both, along meta path $P \leftarrow PAP \rightarrow P$. This could be considered a generalization of the co-citation measure, and capture subtleties such as author affinity to a particular set of papers.

5.4 Path Length Performance

We run several experiments to verify the efficiency improvement achieved using commuting matrices on long meta paths. In Figure 5.1a, we show the performance of the proposed commuting matrix-based approach to compute AsymSim. For two base meta paths, APA and $AP \rightarrow PA$, we show the increase in running time for computing AsymSim for $(APA)^2$ and $(AP \rightarrow PA)^2$. As we see in Figure 5.1a, the running time for directly computing these longer paths increases dramatically, while the running time increases only slightly using the commuting matrix approach. We gain this performance advantage by computing the adjacency matrix of the base meta path, performing matrix multiplication to get the adjacency matrix for the full meta path. Figure 5.1b shows the breakdown of running time between computing these partial adjacency matrices and the matrix multiplication for each example. As shown here, computing longer paths increases the running time only by the time for the additional matrix multiplications, and so with repetitions of the base meta path, the total time increases linearly. In practice, we may compute these partial adjacency matrices offline, so that only the matrix multiplication needs to be performed at query time.



(a) Time for Direct Computation Versus Using Commuting Matrices



(b) Breakdown of Commuting Matrix Computation

Figure 5.1: Commuting Matrices Performance for AsymSim

Chapter 6

Discussions

AsymSim has several properties ideal for a similarity measure. These properties are formally shown in [9], but we outline the similar argument for two properties in brief:

1. **Symmetric**: AsymSim is defined only on symmetric meta paths P, where $P = P^{-1}$. Thus, $s(x, y, P) = s(x, y, P^{-1}) = s(y, x, P)$, and so AsymSim is a symmetric measure.

2. **Self-Maximum**: In the AsymSim computation, since the number of paths to meta-neighbors of each object x (incoming path instances of Q for $P = Q^{-1}Q$) is always greater than or equal to the number of those paths of P that connect through the meta-neighbors to y, we know that the AsymSim measure will always be at most 1. Since the measure is based on path counts, the value is always at least 0, and since the number of paths through meta-neighbors along Q^{-1} is equivalent to the sum of the number of meta paths through each individual neighbor, AsymSim is exactly 1 for any $s(x, x, P)$. Therefore, $s(x, x, P) \in [0, 1]$ for any object x or meta path P.

In our work, we focus on peer similarity using a single meta path, although in practice, multiple meta paths may be weighted together to construct a more fine-grained similarity measure for particular applications. *AsymSim* can be easily used in this framework, by simply weighting our similarity scores across various meta paths. We choose to focus on in-neighbors for the nodes involved in the similarity measure. We believe that this is intuitive and simpler than balancing a measure between in and out neighbors, but also realize that out-neighbors may play a role in similarity semantics that is overlooked by our current method.

AsymSim and PathSim measures capture only the local structure of the network. These measures are not well-suited to similarity globally in the network, such as finding authors with the most similar *citation count* in the network, irrespective of the similarity of their work. These types of similarity measures carry important semantic meaning, but further work needs to be done to extend the meta path-based framework to handle these measures, perhaps by generalizing the computation using the meta path commuting matrix.

Chapter 7

Conclusion

Similarity search is a fundamental problem in data mining, and for heterogeneous networks, this is particularly true. Peer similarity search on heterogeneous information networks contains subtle semantics, and users may want to adjust these semantics while using the same similarity measure. Previous work such as PathSim [9] addresses this problem by introducing a framework that allows users to define similarity semantics using meta paths, or paths traversing a particular series of object and relation types. PathSim is only defined on meta paths containing symmetric relations, but asymmetric relations may be valuable for peer similarity. In DBLP, ignoring asymmetric relations means ignoring citations, an important relation between papers that helps establish the 'reputation' of documents, authors, and conferences.

We propose AsymSim, a peer similarity measure that handles both symmetric and asymmetric relations in information networks, while capturing the same desirable semantics shown in existing work, such as heterogeneous types and balance of visibility. We introduce the concept of meta neighbors to the meta path-based framework, and show that AsymSim captures the semantics of the peer similarity search problem in a more intuitive and descriptive way than existing methods. We demonstrate this improvement through case studies on the DBLP dataset, and show that AsymSim can be computed efficiently using adjacency matrices for path instances in the network, called commuting matrices.

References

[1] Heasoo Hwang. Objectrank: a system for authority-based search on databases. In *In SIGMOD 06: Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 796–798. ACM Press, 2006.

[2] Glen Jeh and Jennifer Widom. Simrank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 538–543, New York, NY, USA, 2002. ACM.

[3] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *Proceedings of the 12th international conference on World Wide Web*, WWW '03, pages 271–279, New York, NY, USA, 2003. ACM.

[4] Zaiqing Nie, Yuanzhi Zhang, Ji rong Wen, and Wei ying Ma. Object-level ranking: Bringing order to web objects. In *Study of the eXplicit Control Protocol (XCP). IEEE Infocom*, 2005.

[5] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[6] Yizhou Sun, Rick Barber, Manish Gupta, Charu C. Aggarwal, and Jiawei Han. Co-author relationship prediction in heterogeneous bibliographic networks. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '11, pages 121–128, Washington, DC, USA, 2011. IEEE Computer Society.

[7] Yizhou Sun and Jiawei Han. *Mining Heterogeneous Information Networks: Principles and Methodologies*. Morgan & Claypool, 2012.

[8] Yizhou Sun, Jiawei Han, Jing Gao, and Yintao Yu. itopicmodel: Information network-integrated topic modeling. In *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, ICDM '09, pages 493–502, Washington, DC, USA, 2009. IEEE Computer Society.

[9] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S. Yu, and Tianyi Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. In *In VLDB' 11*, 2011.

[10] Jie Tang, Limin Yao, Duo Zhang, and Jing Zhang. A combination approach to web user profiling. *ACM TKDD*, 5(1):1–44, 2010.

[11] Jie Tang, Duo Zhang, and Limin Yao. Social network extraction of academic researchers. In *ICDM'07*, pages 292–301, 2007.

[12] Jie Tang, Jing Zhang, Ruoming Jin, Zi Yang, Keke Cai, Li Zhang, and Zhong Su. Topic level expertise search over heterogeneous networks. *Machine Learning Journal*, 82(2):211–237, 2011.

[13] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: Extraction and mining of academic social networks. In *KDD'08*, pages 990–998, 2008.

Appendix A

Author Case Studies Figures

This appendix shows additional results of similarity case studies for particular authors in DBLP.

A.1 'Rakesh Agrawal' Case Study

In our dataset, 'Rakesh Agrawal' is the highest cited author, and thus as an extreme example in the dataset, is an interesting case study to consider for the similarity measures we study.

A.2 'AnHai Doan' Case Study

'AnHai Doan' is an interesting case study in the DBLP network, since he is one of the less visible authors, and his citation count is not exceptionally higher than his publication count.

A.3 'Philip S. Yu' Case Study

In the DBLP network, 'Philip S. Yu' is one of the most highly cited and published authors, but is not an extreme example as we saw with 'Rakesh Agrawal'.

Rank	Author	Citations	Publications	PathSim Score
1	**Rakesh Agrawal**	**1511**	**105**	**1**
2	Hector Garcia-Molina	613	98	0.958
3	H. V. Jagadish	546	106	0.942
4	Nick Koudas	351	79	0.935
5	Jeffrey F. Naughton	801	85	0.927
6	Surajit Chaudhuri	581	96	0.927
7	Divesh Srivastava	464	103	0.924
8	Michael Stonebraker	463	74	0.922
9	Raghu Ramakrishnan	652	95	0.912
10	Christos Faloutsos	634	127	0.901

Table A.1: Similarity to *Rakesh Agrawal* Using *PathSim* with $APCPA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Rakesh Agrawal**	**1511**	**105**	**1**
2	Ramakrishnan Srikant	696	34	0.746
3	Jiawei Han	665	168	0.701
4	Philip S. Yu	556	216	0.573
5	Raghu Ramakrishnan	652	95	0.482
6	Arun N. Swami	404	15	0.454
7	Rajeev Motwani	542	36	0.423
8	Jian Pei	270	70	0.418
9	Tomasz Imielinski	343	30	0.379
10	Charu C. Aggarwal	238	65	0.326

Table A.2: Similarity to *Rakesh Agrawal* Using *AsymSim* with $AP \leftarrow PAP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Rakesh Agrawal**	**1511**	**105**	**1**
2	Jiawei Han	665	168	0.771
3	Ramakrishnan Srikant	696	34	0.753
4	Christos Faloutsos	634	127	0.683
5	David J. DeWitt	876	74	0.671
6	Philip S. Yu	556	216	0.671
7	Jeffrey F. Naughton	801	85	0.656
8	Hans-Peter Kriegel	617	102	0.653
9	Jennifer Widom	726	64	0.643
10	Raghu Ramakrishnan	652	95	0.639

Table A.3: Similarity to *Rakesh Agrawal* Using *AsymSim* with $AP \leftarrow PCP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Rakesh Agrawal**	**1511**	**105**	**1**
2	Ramakrishnan Srikant	696	34	0.763
3	Jiawei Han	665	168	0.656
4	Arun N. Swami	404	15	0.619
5	Philip S. Yu	556	216	0.547
6	Rajeev Motwani	542	36	0.525
7	Tomasz Imielinski	343	30	0.504
8	Jeffrey F. Naughton	801	85	0.496
9	David J. DeWitt	876	74	0.494
10	Jennifer Widom	726	64	0.480

Table A.4: Similarity to *Rakesh Agrawal* Using *AsymSim* with $AP \leftarrow PTP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Rakesh Agrawal**	**1511**	**105**	**1**
2	Ramakrishnan Srikant	696	34	0.690
3	Arun N. Swami	404	15	0.395
4	Tomasz Imielinski	343	30	0.362
5	Jiawei Han	665	168	0.344
6	Philip S. Yu	556	216	0.221
7	Raghu Ramakrishnan	652	95	0.204
8	Jeffrey D. Ullman	438	43	0.191
9	Rajeev Motwani	542	36	0.190
10	Jian Pei	270	70	0.188

Table A.5: Similarity to *Rakesh Agrawal* Using *AsymSim* with $AP \leftarrow P \rightarrow PA$ meta path

Rank	Author	Citations	Publications	PathSim Score
1	**AnHai Doan**	**118**	**26**	**1**
2	Jignesh M. Patel	92	25	0.976
3	Xuemin Lin	53	25	0.961
4	Balakrishna R. Iyer	122	23	0.957
5	Jayant R. Haritsa	44	35	0.955
6	Jun Yang	93	34	0.948
7	Walid G. Aref	63	33	0.947
8	Ming-Chien Shan	38	23	0.941
9	Won Kim	162	30	0.938
10	Anastassia Ailamaki	71	25	0.938

Table A.6: Similarity to *AnHai Doan* Using *PathSim* with $APCPA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**AnHai Doan**	**118**	**26**	**1**
2	Jayant Madhavan	98	10	0.769
3	Alon Y. Halevy	236	34	0.718
4	Erhard Rahm	89	16	0.532
5	Philip A. Bernstein	171	44	0.436
6	Bin He	41	10	0.421
7	Pedro Domingos	291	47	0.410
8	Kevin Chen-Chuan Chang	112	32	0.388
9	Renée J. Miller	152	29	0.366
10	Joann J. Ordille	83	4	0.343

Table A.7: Similarity to *AnHai Doan* Using *AsymSim* with $AP \leftarrow PAP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**AnHai Doan**	**118**	**26**	**1**
2	Moni Naor	122	3	0.978
3	Anant Jhingran	92	14	0.977
4	Kevin Chen-Chuan Chang	112	32	0.973
5	Stefano Ceri	96	36	0.957
6	Krithi Ramamritham	114	41	0.956
7	Venkatesh Ganti	124	21	0.955
8	Qiong Luo	109	19	0.954
9	Erhard Rahm	89	16	0.952
10	Narain H. Gehani	80	14	0.941

Table A.8: Similarity to *AnHai Doan* Using *AsymSim* with $AP \leftarrow PCP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**AnHai Doan**	**118**	**26**	**1**
2	Jayant Madhavan	98	10	0.931
3	Erhard Rahm	89	16	0.787
4	Alon Y. Halevy	236	34	0.747
5	Philip A. Bernstein	171	44	0.736
6	Joann J. Ordille	83	4	0.678
7	Susan B. Davidson	106	24	0.669
8	Bin He	41	10	0.663
9	Mauricio A. Hernández	119	13	0.660
10	Renée J. Miller	152	29	0.644

Table A.9: Similarity to *AnHai Doan* Using *AsymSim* with $AP \leftarrow PTP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**AnHai Doan**	**118**	**26**	**1**
2	Jayant Madhavan	98	10	0.718
3	Alon Y. Halevy	236	34	0.593
4	Erhard Rahm	89	16	0.451
5	Philip A. Bernstein	171	44	0.388
6	Pedro Domingos	291	47	0.374
7	Bin He	41	10	0.320
8	Renée J. Miller	152	29	0.265
9	Kevin Chen-Chuan Chang	112	32	0.257
10	Jaewoo Kang	25	7	0.225

Table A.10: Similarity to *AnHai Doan* Using *AsymSim* with $AP \leftarrow P \rightarrow PA$ meta path

Rank	Author	Citations	Publications	PathSim Score
1	**Philip S. Yu**	**556**	**216**	**1**
2	Jiawei Han	665	168	0.920
3	Christos Faloutsos	634	127	0.759
4	Hans-Peter Kriegel	617	102	0.689
5	Wei Wang	135	83	0.679
6	Rakesh Agrawal	1511	105	0.645
7	Divesh Srivastava	464	103	0.639
8	Beng Chin Ooi	130	70	0.606
9	Hector Garcia-Molina	613	98	0.593
10	Nick Koudas	351	79	0.592

Table A.11: Similarity to *Philip S. Yu* Using *PathSim* with $APCPA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Philip S. Yu**	**556**	**216**	**1**
2	Charu C. Aggarwal	238	65	0.618
3	Rakesh Agrawal	1511	105	0.573
4	Ramakrishnan Srikant	696	34	0.565
5	Jiawei Han	665	168	0.554
6	Arun N. Swami	404	15	0.395
7	Johannes Gehrke	320	56	0.390
8	Jian Pei	270	70	0.383
9	Rajeev Motwani	542	36	0.382
10	Raghu Ramakrishnan	652	95	0.350

Table A.12: Similarity to *Philip S. Yu* Using *AsymSim* with $AP \leftarrow PAP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Philip S. Yu**	**556**	**216**	**1**
2	Jiawei Han	665	168	0.946
3	Ramakrishnan Srikant	696	34	0.941
4	Arun N. Swami	404	15	0.899
5	Christos Faloutsos	634	127	0.893
6	Rajeev Motwani	542	36	0.884
7	Tomasz Imielinski	343	30	0.843
8	Johannes Gehrke	320	56	0.842
9	Hans-Peter Kriegel	617	102	0.831
10	Jian Pei	270	70	0.798

Table A.13: Similarity to *Philip S. Yu* Using *AsymSim* with $AP \leftarrow PCP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Philip S. Yu**	**556**	**216**	**1**
2	Jiawei Han	665	168	0.802
3	Arun N. Swami	404	15	0.713
4	Rajeev Motwani	542	36	0.709
5	Raghu Ramakrishnan	652	95	0.706
6	Johannes Gehrke	320	56	0.704
7	Ramakrishnan Srikant	696	34	0.704
8	Dimitrios Gunopulos	264	46	0.687
9	Miron Livny	277	35	0.673
10	Tomasz Imielinski	343	30	0.668

Table A.14: Similarity to *Philip S. Yu* Using *AsymSim* with $AP \leftarrow PTP \rightarrow PA$ meta path

Rank	Author	Citations	Publications	AsymSim Score
1	**Philip S. Yu**	**556**	**216**	**1**
2	Charu C. Aggarwal	238	65	0.449
3	Ming-Syan Chen	116	61	0.332
4	Jiawei Han	665	168	0.291
5	Haixun Wang	100	67	0.273
6	Jong Soo Park	94	4	0.255
7	Joel L. Wolf	71	8	0.228
8	Rakesh Agrawal	1511	105	0.221
9	Ramakrishnan Srikant	696	34	0.185
10	Johannes Gehrke	320	56	0.182

Table A.15: Similarity to *Philip S. Yu* Using *AsymSim* with $AP \leftarrow P \rightarrow PA$ meta path